IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Journal Européen des Systèmes Automatisés
Vol. 55, No. 5, October, 2022, pp. 679-685
Journal homepage: http://iieta.org/journals/jesa

Simulation of Reinforcement Learning Algorithm for Motion Control of an Autonomous Humanoid

Deepak Bharadwaj*, Durga Dutt

Mechanical Cluster, UPES Dehradun, Dehradun 248001, India

Corresponding Author Email: dbharadwaj@ddn.upes.ac.in

https://doi.org/10.18280/jesa.550514

Received: 25 July 2022
Accepted: 7 October 2022

Keywords:
Markov decision process (MDP), reinforcement learning agents (RL), transition probabilities, reward

ABSTRACT

Autonomy is an issue in robotics systems. Currently the robotics research community is focusing on autonomy in the decision. The pre-programmed humanoid robots perform the operation in a known scenario. The presence of an obscure environment, a lack of awareness of the environment, humanoid robot fails to perform the random task. In such cases, the preprogrammed robot needs to be reprogrammed to enable it to perform in the changing environment. There is very limited success achieved in the autonomy of the decision-making process. This research work considered the problem of an autonomy while the deicion making in th real time interaction. The reinforcement algorithm helps to do the task in such type of unstructured and unknown environment. Reinforcement learning problems are categorized into partial Markov Decision Processes (MDP). The goal of RL agent is to minimize its immediate and expected costs. When the system interacts with the Markov Decision Process, RL agent passes through an intermediate sequence of states that depends on another by transition probabilities. The agents action takes, and the agents experience a sequence of immediate costs incurred. Reinforcement Learning and teaching approach like Queue Learning (Q-Learning) is implemented for humanoid robot for navigation and exploration. The Q-learning expresses the expected costs to go of a state action pair defined, which is meant to express the expected costs arising after having taken action in the state following policy. Based on the optimal policy of the reinforcement algorithm, a reinforcement controller was implemented. The transition probabilities of the controller depend on the randomness of the controller. The random values of the controller decide the action. Simulations were carried out for the different positions of the proposed model, and an interesting result were was observed while the transition from the sitting position to the goal position.

1. INTRODUCTION

From last two decade, the research community focus on the developing of the human like walking machine. These walking machine do the action similar to humans [1]. Anthropomorphic robot has ability to gather the information from the surrounding and can perform the desired task. The most challenging problem in Anthropomorphic Humanoid robot is the decision making, perception, interaction to the environment and locomotion. The development of actuator and sensoric minimized the locomotion problem, but still totally locomotion problem is not solved. Decision making and perception is still a big challenge to the robotic research community.

The highly specialized robot is implemented in the industry for mass production. Due to human-like shape, the humanoid robot is the best suited for acting in our everyday environment like stairs climbing, door closing and opening handles, tools and human-centered designs. Humanoid robot cannot perform the task in a dynamic scenarios. Human like walking machine must learn from its mistakes and will adapt accordingly without any help from the other guidance. Force sensing and compliance at each humanoid robot joint can allow the robot to safely act in the unknown environment. The implementation of humanoid robot to the community is always an economic and social issues.

The principle motive behind the designing of the humanoid robot is to ease human efforts and do the jobs on behalf of them. The development of advanced and humanoid robotics has certain impact on industrial growth and social impact. The futuristic humanoid robots will be able to do industrial and no-industrial work. Humans are eager to project emotion into machines and replicate themselves into mechanical form.

1.1 Motivation

The child suffering from the autism spectrum disorder disease deficit in social interaction and communication to real world. Similarly, the old age people unable to do the household activity in the home alone environment. In such type of scenario, a walking machine humanoid robot plays an important role to the social life of old age people and child suffering from the autism disorder disease.

The behavioral based interaction and self-decision-making humanoid robot recognize the eye's glaze and behavior of the old age people and child. After recognizing these parameters, humanoid robot executes the task for the people suffering from physical and mental disorder disease. Till today, partial human thinking behavior implemented in the humanoid robot.

Human decision making like intelligence pattern to be implemented in the robotic manipulator is the motivation behind this work. So that, it can be useful for the physically challenged people.

2. REINFORCEMENT LEARNING ALGORITHM

The reinforcement algorithm applied in assistive robot for educational application. The child's gaze provides the information to the robot. The reinforcement algorithm has a set of state. State is the dimensional features. The action has finite discrete set of action and generate set of actions for the different state. The Q-learning rule helps to choose the action depending upon the task. After choosing the action, the transition takes place and a reward associated to the action and learns from the past history. The reinforcement algorithm decomposes the task into set of discrete action, so that it can be easily understood by children-robot interaction [2]. The reinforcement learning accumulated the knowledge from the dynamic balance of the humanoid robot and improving the gait during walking. The control architecture of gait synthesizer has three components. The neural network trains the action selection network using the error signal received from the external reinforcement. For the desired state, the action evaluation maps a state and a failure into a scalar score. The stochastic action modifier use the recommendation action and reinforcement to produce a dynamic walking [3, 4]. The work on episodic reinforcement learning to control the motor primitives in dynamic situation. The policy gradient method used in reinforcement algorithm. The motor primitives described as two coupled differential equations i.e. a canonical system with movement phase and possible external coupling. For the desired control of motor both the dynamics of system are choose for stable condition. The deterministic mean policy depends on the joint position and the basis function. The basis function is the motor primitive parameters [5].

A collaborative interaction between human and robot based on reinforcement learning. The learning is based on collaborative Q-learning approach and provides the robot to self-awareness and autonomy. In collaborative Q learning algorithm, there is two levels of collaboration between human and robot. In the first level, the robot decides the action and update its state-action values. In second level of collaboration, robot takes the request from human advisors. Robot is switching from the autonomous mode to semi-autonomous mode based on the polices [6]. The reinforcement learning algorithm for humanoid gait optimization. The actor-critic learning applied for the experience replay and fixed-point method to determine the step size. The Markov decision process provides the solution for the reinforcement algorithm to control the humanoid robot gait. The control process of actor-critic works in discrete time to select the state and select the proper action. The transition between current states to next state happens and a reward assigned to state and action. The stochastic control and value function update the learning parameter based on the data collected [7]. The fuzzy reinforcement hybrid control algorithm for the bipedal robot locomotion. The controller has two feedback loops around the zero-moment point. The centralized dynamic controller keeps tracking of the robot's normal trajectory and a fuzzy reinforcement feedback compensate the dynamic reactions of the ground around the zero-moment point. The fuzzy reinforcement control algorithm structure based on the actor critic temporal difference method. The policy represents the set of control parameters [8, 9].

3. REINFORCEMENT LEARNING CONTROLLER

The reinforcement learning which could control the iCub humanoid robot. iCub learns a world model from experience and controlling the actual hardware in real time with some restrictions. Reinforcement learning discretize the real configuration of the robot in configuration space. The modular behavior environment of iCub humanoid robot generate the action and robot try to go in the transition state. The Markov model develop the path planner and connect the state to the near state [10].

The deep reinforcement learning algorithm to train the control policies for the humanoid robot interactions. The control problem is formalized from the Markov decision process. The input to the control policy is, joint position, velocity and sensor reading of the hand. The motion capture system captures the position of the leg. The positive reward is given as 1 when there is a proper switching of leg from one position to another position. Otherwise, negative reward been assigned to the reinforcement controller. The output of the control policy actuates the humanoid arm. The reward is provided to correct end configuration of the humanoid arm [11].

The reinforcement learning using Bayesian optimization improve the whole body motion control. The Bayesian optimization is a nonlinear and nonconvex optimization technique. It evaluates the cost function in the robotics and optimize the set of parameters. To ensure smooth trajectory, the whole-body control guided by the task in a series of waypoints. Three components of costar are evaluated for the execution of task. The optimization variables selected from the trajectory waypoint [12].

A model-based reinforcement algorithm with decision tree to train the humanoid robot to kick goals. The model-based reinforcement algorithm, learning takes place aggressively during model learning. The Q-learning approach adopted for the model free reinforcement learning. The Q-learning update the state -action for every state -action pair. The reinforcement learning with decision tree take the action with a highest value and entering into a new state. After entering into a new state, award will be received in the new state. Observing new experience through the model, the algorithm updates the parameter through the model [13]. The application of batch reinforcement learning in challenging and crucial domain. Reinforcement learning help the robot to gain the ideas form the repetitive interaction from the environment. The batch reinforcement control algorithm consists of sampling experience, training and batch supervised learning. The training pattern set estimates the value function. The batch supervised generates new estimate for the value function form the training set pattern. The behavior-based approach used to implement the reinforcement algorithm to take the decision [14]. The batch reinforcement requires the sampling data and not able to take the decision in unknown environment. In the Reinforcement algorithm, the robot is self-capable to handle the situation.

The adaptive allocation method for reinforcement control algorithm for humanoid motion control. The actor critic learning adopted for the reinforcement learning. This method has a separate memory to represent the policy i.e. independent

from the value function. The actor calculates the action value for the humanoid robot when it observes the state in the environment. The critic receives the reward and provide the temporal difference. The learning is simulated on the virtual body of the humanoid robot to stand up from a chair. The humanoid observes the wait, knee, ankle and pitch angle of body. The humanoid robot learns to fall down backward. Afterwards it falls down forward. Finally, it stands up and control its body [15]. The dynamic control approach for the humanoid bipedal walking. The controller involves two feedback loops. The computational torque controller receives the input from impact force controller and reinforcement controller. The reinforcement controller maintains the torso movement with the help fuzzy feedback. The policy gradient reinforcement learning control the trajectory of dynamic walking of the humanoid robot [16]. The actor critic neural network architecture for continuous action policy of reinforcement learning. The deep deterministic policy gradient method controls the humanoid body control task. The deterministic policy developed by the actor network. The action value generates by the critic network. The temporal difference minimized by the training of critic network. The immediate reward received upon the action and update the learning parameter to the database of control architecture [17]. The different approach for reinforcement learning algorithm for humanoid robot. The natural actor critic learning control the motor of humanoid robot. The movement plan has set of joint position and joint velocity of the humanoid robot. The system has point-to-point continuous movement i.e. the episodic task of the reinforcement algorithm. The evaluation of basis function for the value is done by the actor critic network [18].

Algorithm for programming robot by demonstration. When unexpected perturbations occur, robot is unable to perform the task. The reach of constrained task to the robot by a learned speed trajectory. When the feet come into the interaction of ground surface, at that time environment is fixed, but the same feet come into the contact of the snow like ground, the environment is different. The natural actor critic network evaluates the policy by approximating the state action values. The simulation is carried out for the cubic box and obstacle. Using the reinforcement learning, the system takes 330 trials to achieve the goal [19]. The control policies in simulation that can transfer to dynamical physical system. The policy gradient learning method used in reinforcement algorithm to optimize the parameters. The natural policy gradient algorithm pushing the task to learn. The training of the policy determines the action to take and gain a good reward. The structure of training informs the policy behavior with the time required to execute the task. The reward function reduces the gap between the robot and the target [20]. The reinforcement algorithm which maps the circumstances to meta parameters. The motor primitives used for the meta parameters learning. The dynamical movement of the motor represented in the first order differential equation for the critical damped. The goal parameter is the function of the amplitude parameter represents the complex movement. All degree of freedom of the system synchronize in the dynamical equation in the canonical form [21]. A reinforcement learning algorithm to optimize the parameter values for the generation of gait pattern in humanoid robot. Locomotion control achieve by the central pattern generator. The three-oscillator attached in the foot of the humanoid robot. Each oscillator has six sub oscillators related to the axis and configured to the parameters. The parameters are divided into three groups of offset parameters, oscillation parameter and feedback parameters. Two parameters have selected for the optimize the gait [22]. The intrinsic interactive reinforcement learning algorithm for human robot interaction based on the gesture posture. The human electroencephalogram generated feedback used for the reward. The leap motion controller recognizes the human gesture to learn the robot and parallelly the robot maps the gesture for action. The contextual bandit approach used to enable the robot's action provided by the human gestures [23].

3.1 Reinforcement model

The reinforcement learning model depends on the discrete set of environment states S, discrete set of agent action A and set of reward signal {0, 1}. Trial and error search and the delayed reward are the two characteristics of reinforcement model. The RL model is defined by characterizing a learning problem. Figure 1 shows the model of the reinforcement model.

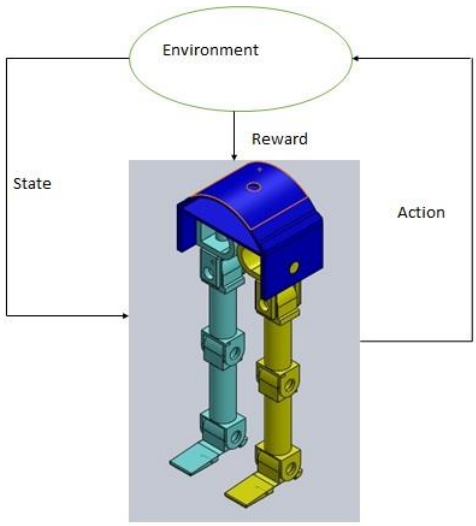

Figure 1. Reinforcement model of lower body

The basic model of Reinforcement learning consists of three steps as follows:

Step 1: The agent of reinforcement model sense an input *i* from the current state of the surrounding and environment.

Step 2: Agent chooses the action from the set of actions (A) like switch to next state, switch to previous state and idle stop.

Step 3: The transition of state is sending the information through a scalar reinforcement signal (r).

3.2 Q–Learning method

Learning from the environment is very complicate to human like manipulator. Supervised learning and unsupervised learning are not useful in such scenario. Reinforcement learning is different from these two types of learning. Reinforcement learning has model-based reinforce learning and model free reinforcement learning approach. Model based reinforcement learning has limitation, it cannot adopt the system changes and fail to do the task. In the model free reinforcement learning, system adaptability is very high and self-capable to take the decision. Q-learning is adopted for the model free reinforcement model due to its simplicity and online learning. Reinforcement learning is model free learning,

it doesn't depend on the internal parameters. Q-learning is one of approximate tool that is being used for targeting the goal position. It is approximate programming of Markov Decision Process. The agent has to learn navigation strategy from the environment through a reward signal and generate a path. The path generated by the agent must be shortest and collision free path. The Q-learning approach makes necessary changes in: heuristic search. The search algorithm finds the way of shortest path [24].

In this approach of reinforcement learning from demonstration, the humanoid robot learns a reward function from the demonstration and a task model from repeated attempts (trials) to perform the task [25].

The execution of the Q-learning algorithm is as follows:

Step 1: Initialization of $Q(s, a)$ to generated random values by processor.

Step 2: Observing the current state, s.

Step 3: Based on the random values, an action, a, will perform.

Step 4: Observing the switching state, s', and awarding the reward, r.

Step 5: $Q(s, a) \leftarrow (1-\alpha)\ Q(s, a)+\alpha(r+\gamma max_{a'} Q(s', a'))$.

3.3 Implementation Q-learning algorithm in MATALB

Based on the Q-learning algorithm, one of the sample code will be developed in the MATLAB as shown below. $Q(s, a)$, represents the state-action pair, α represents epsilon, γ represents learning rate. R is the reward. α represents the iteration of the individual joint movement while reaching to set point to the target point. The significance of the γ represents the ability to learn to reach the target point accurately.

```
learnRate
γ =.90
epsilon
α =.6
Decay of the epsilon
Epsilon_Decay=.8
Discount=.9
Success_Rate=1;
Achieved the goal point
Win_Bonus =100
Input sense for sensor
Start_Pt=[-40]
Goal Point Reached
Goal_Pt=[0]
Max_Epi=50000;
Trajctyory Action
Action=1°
Actions=[0, Action]
Intermediate state
x=linspace(Start_Pt, Goal_Pt,10)
Lngth of states
States=zeros(length(x),1)
Index=1;
For j=1:length(x)
States(index,1)=x(j)
Index=Index+1;
EnD
Assignmnt of reward
R=States*.1
Q Values for the state and action
Q=repmat(R,[1,3
Z_1=Start_Pt
For Episodes=1:Max_Epi
[~,SI_dx]=Min(Sum(States-
repmat(Z_1,[Size(States,1),1])).^2,2))
Picking
if(rand()epsilon!!episodes=Max_Epi)&&rand()<=Success_R
ate)
Best Action
[~,aIdx]=max(Q(SI_dx,:)
else
aIdx=randi(length(actions),1)
end
T=actions(aIdx)
Z_2=Z_1+T
Z_1=Z_2
if(Z_2=Goal_Pt)
success=true
Bonus=Win_Bonus
[~,snewIdx]=min(sum(States-
repmat(z1),[size(States,1),1])).^2,2))
Q(SI_dx,aIdx)=Q(SI_dx,aIdx)+learnRate*(R(Snew_Idx)+Dis
count*max(Q(Snew_Idx,:) )-Q(SI_dx,aIdx)+Bonus)
Break
Else
Bonus=0
Success=False
End
```

3.4 Limitation of reinforcement learning

The most challenges are the trade-off exploration and exploitation in the reinforcement learning. An agent has to erase the previous learning from the past and has to make precise selections of action in the future. The agent of reinforcement learning is independent and determined its own by learning from the interaction to the environment. The learning rate of agent can improve the decision-making process while interacting to the environment. The agent considered the environment. The intermediate states and features are the parameters of the environment. Agent sense the environment and learns the optimal policy while transition from the initial sate to the goal state by taking the action in each state. The agent must be aware of the states while interacting with the environment. An agent learns from reinforcement feedback received from its environment known as positive reward and negative reward. The agents maximize the positive reward or minimize the negative reward. The hardware limit stops like limit switch, limit sensor avoids the exploration and exploitation problem. Actions affect the intermediate state of the system and rewards and have the ability to optimize the system's state. Continuous learning and adapting through interaction with environment help the agent to learn online in terms of performing the required task and improving its behavior in real time [26].

4. REINFORCEMENT CONTROLLER IMPLEMENTATION

A Simulink block has been created for the reinforcement controller in the MATLAB as shown in Figure 2.

The decision taken by the lower body of humanoid robot by knowing the current state from the environment. The reinforcement controller does not involve any kinematic and

dynamic description. The switching will happen its own. One of the states start from the 0° of ankle joint, -10° of knee joint and -20° of the hip joint for the right leg. The left leg start forms the same and opposite to the right leg joint values. These values were sensed by the agent from the environment with the help of sensor. The goal state of the system reaches to the 10°, 15° and 20° of the ankle, knee and hip joint respectively. Time elapsed to reach this goal position are independent from the kinematics and dynamics calculation. In reinforcement controller time taken to reach next state, it is totally dependent on the processor capabilities and generating the control signal. The reinforcement controller controls the intermediate position of the joint between the start point and the target point. Figure 3 shows the interaction of reinforcement controller to the lower body of Humanoid robot.

The randomness of the controller helps to take the action. The Q-learning algorithm allows the robot to go from one state to another state. Randomness only contributes to take the action like switch to next position, switch to previous position or stay at same position. Randomness effect the processor execution time, so that there may be delay to reach to target position.

The reward is gain during the switching by the state-action pair. After several running, the state -action pair of the reinforcement controller keeps the updated values. In the next time of running the system executes with the old values and try to switch to the next state.

5. RESULT AND DISCUSSIONS

The simulation experiment was carried out for the reinforcement controller. Figure 4 shows the initial position of lower body of humanoid robot. The initial position of joint 1 is taken as -10°, initial position of joint 4 is taken as 15° and the initial position of joint 5 is -20°. Due to software constraint, the initial position is taken as 0° for all the joint. The initial position of the lower body is vertically upward position and assuming no deviation in other joint 2 and joint 3. Figure 4 shows the initial posture of the lower body.

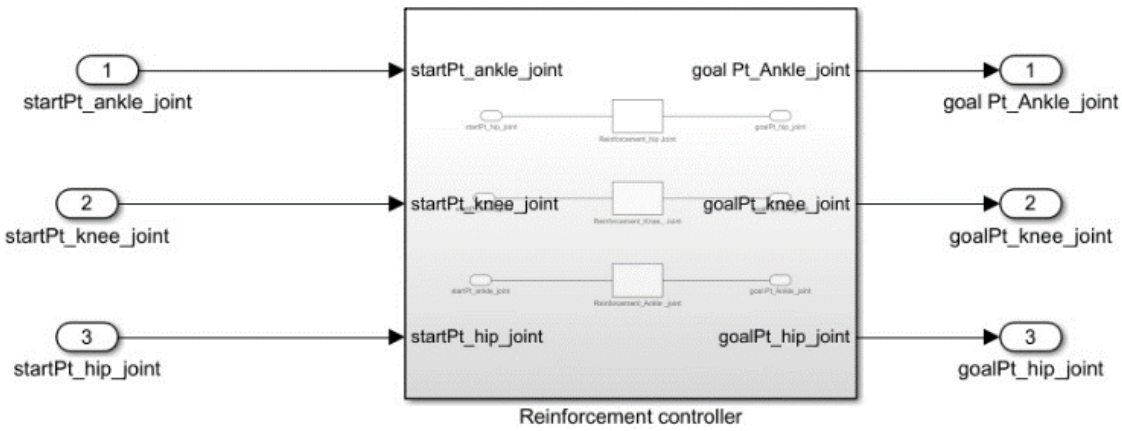

Figure 2. Simulink block diagram of reinforcement controller

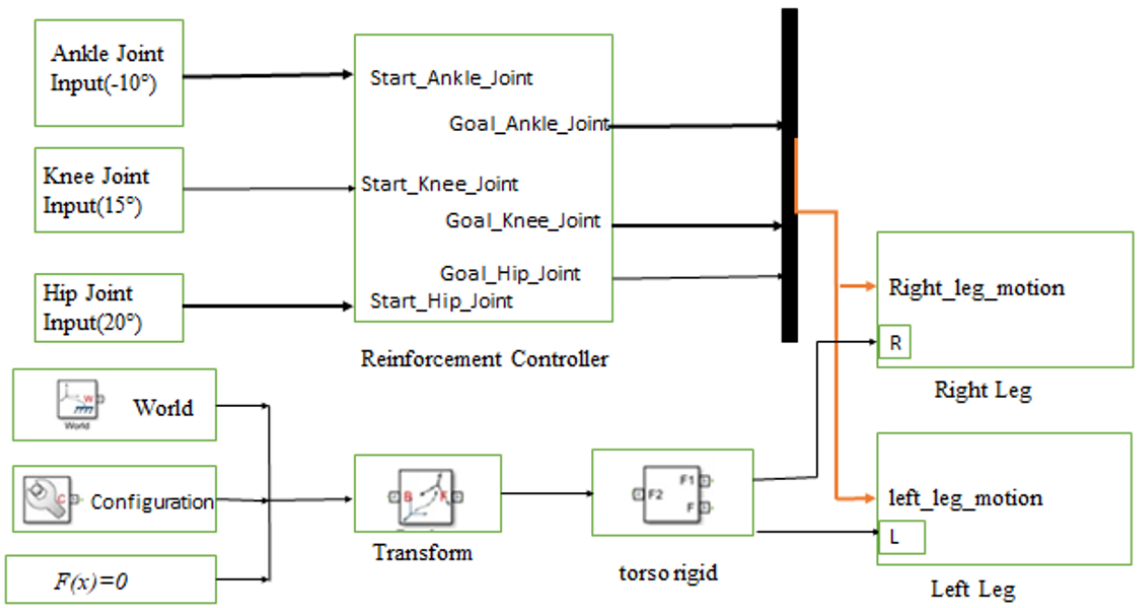

Figure 3. Interfacing of reinforcement controller to lower body of Humanoid robot

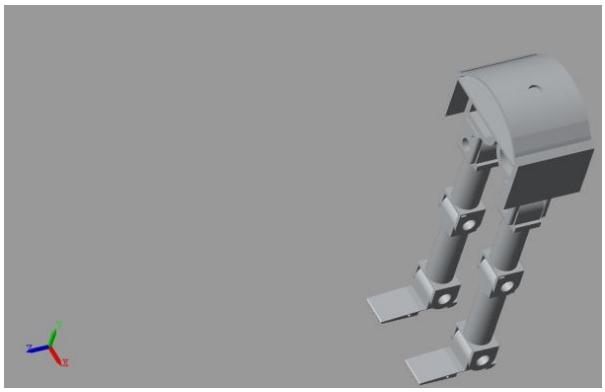

Figure 4. Initial state of lower body of humanoid robot

Reinforcement controller takes the input of joint position of lower body of the humanoid robot as a starting point. The goal point of the joint 1 is 10°, the goal point of joint 4 is -120° and the goal point of joint 4 is 0°. These goal points decided the one of the states of lower body. The controller takes the input and executes the algorithm.

Figure 5 shows the next state of lean of the lower body of humanoid robot.

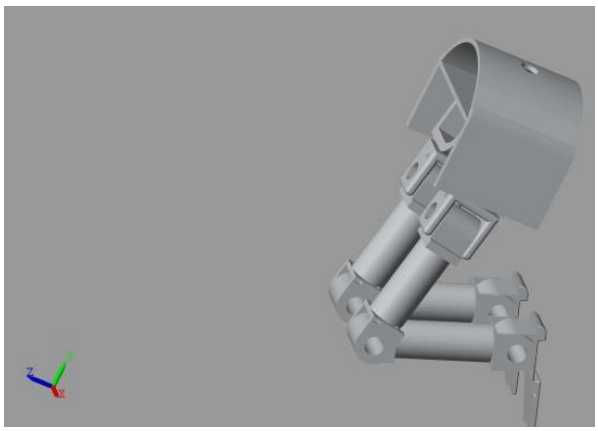

Figure 5. Next state of lower body of humanoid robot

During this transition from one state to next state, there is no kinematics and dynamics involved. Controller takes its own decision to switch over. There is no control over the motion of joint by any user or human interface. The switching is totally depending on the controller decision. The reinforcement controller does not depend upon the preprogram of the robot. It is totally dependent upon the transition of probabilities.

While implementing the reinforcement controller on the embedded system, the optimal value was opted for designing the hardware. Controller will implement that values to switch the system from one position to the target position. Simulation result helps to developer to choose the different control parameters to switch the system from one to another.

6. CONCLUSIONS

Selef awareness and stability is the main issues in the humnaoid robot. More number of joint and link, the walking machine manipultor bcomes complicated and the reach of the dexterity become less. The existing kinematctic configuaration causes the locomotion problem. The partiaal success has been obtaind towards the self decision making robot. The reinforcement algorithms implemented to take the decison in the unstructed and unknown enevironment. The Q-algorithm adpoted for the developing the set of instructions. The partial Markove Decision process model were considerd for the developing the reinforcement controller. The randomness of the controller deos not depend on the other human assiatance and the user. The reward function in the reinforcement algorithm is allowing to take the action in unknown and unstructrured environment. Still the randomness factor is issues in reinforcement algorithm.

ACKNOWLEDGMENT

This work is supported by the UPES University.

REFERENCES

[1] Kalra, H.K., Chadha, R. (2018). A review study on humanoid robot SOPHIA based on artificial intelligence. International Journal of Technology and Computing (IJTC), 4: 31-33.

[2] Velentzas, G., Tsitsimis, T., Rañó, I., Tzafestas, C., Khamassi, M. (2018). Adaptive reinforcement learning with active state-specific exploration for engagement maximization during simulated child-robot interaction. Paladyn, Journal of Behavioral Robotics, 9(1): 235-253. https://doi.org/10.1515/pjbr-2018-0016

[3] Katić, D., Vukobratović, M. (2003). Intelligent control techniques for humanoid robots. In 2003 European Control Conference (ECC), Cambridge, UK, pp. 1839-1844. https://doi.org/10.23919/ecc.2003.7085233

[4] Bharadwaj, D., Dutt, D. (2021). Design and development of low-level automation for the picking and placing of the object using pneumatic suction. Journal Européen des Systèmes Automatisés, 54(6): 865-870. https://doi.org/10.18280/jesa.540608

[5] Kober, J., Peters, J. (2008). Policy search for motor primitives in robotics. Advances in Neural Information Processing Systems, 21(1): 87-99. https://doi.org/10.2217/1745509X.2.1.87

[6] Kartoun, U., Stern, H., Edan, Y. (2010). A human-robot collaborative reinforcement learning algorithm. Journal of Intelligent & Robotic Systems, 60(2): 217-239. https://doi.org/10.1007/s10846-010-9422-y

[7] Wawrzyński, P. (2012). Autonomous reinforcement learning with experience replay for humanoid gait optimization. Procedia Computer Science, 13: 205-211. https://doi.org/10.1016/j.procs.2012.09.130

[8] Katić, D., Vukobratović, M. (2006). Control algorithm for humanoid walking based on fuzzy reinforcement learning model of the robot's mechanism. SISY 2006 4th Serbian-Hungarian Jt. Symp. Intell. Syst., pp. 81-93.

[9] Bharadwaj, D., Mishra, N., Pathak, M. (2022). Kinematic and singularity analysis of 10 DOF lower body of humanoid robot. Math. Model. Eng. Probl., 9(2): 484-490.

[10] Frank, M., Leitner, J., Stollenga, M., Förster, A., Schmidhuber, J. (2014). Curiosity driven reinforcement learning for motion planning on humanoids. Frontiers in Neurorobotics, 7: 25. https://doi.org/10.3389/fnbot.2013.00025

[11] Christen, S., Stevšić, S., Hilliges, O. (2019).

Demonstration-guided deep reinforcement learning of control policies for dexterous human-robot interaction. In 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, pp. 2161-2167. https://doi.org/10.1109/ICRA.2019.8794065

[12] Lober, R., Padois, V., Sigaud, O. (2016). Efficient reinforcement learning for humanoid whole-body control. In 2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids), pp. 684-689. https://doi.org/10.1109/HUMANOIDS.2016.7803348

[13] Hester, T., Quinlan, M., Stone, P. (2010). Generalized model learning for reinforcement learning on a humanoid robot. In 2010 IEEE International Conference on Robotics and Automation, pp. 2369-2374. https://doi.org/10.1109/ROBOT.2010.5509181

[14] Riedmiller, M., Gabel, T., Hafner, R., Lange, S. (2009). Reinforcement learning for robot soccer. Autonomous Robots, 27(1): 55-73. https://doi.org/10.1007/s10514-009-9120-4

[15] Iida, S., Kato, S., Kuwayama, K., Kunitachi, T., Kanoh, M., Itoh, H. (2004). Humanoid robot control based on reinforcement learning. In Micro-Nanomechatronics and Human Science, 2004 and The Fourth Symposium Micro-Nanomechatronics for Information-Based Society, pp. 353-358. https://doi.org/10.1109/MHS.2004.1421274

[16] Katic, D., Rodic, A., Vukobratovic, M. (2008). Reinforcement learning control algorithm for humanoid robot walking. International Journal of Information & Systems Sciences, 4(2): 256-267.

[17] Danel, M. (2017). Reinforcement learning for humanoid robot control. Poster, 1-5.

[18] Peters, J., Vijayakumar, S., Schaal, S. (2003). Reinforcement learning for humanoid robotics. In Proceedings of the third IEEE-RAS International Conference on Humanoid Robots, Karlsruhe, Germany, Sept. 29-30, pp. 1-20.

[19] Guenter, F., Hersch, M., Calinon, S., Billard, A. (2007). Reinforcement learning for imitating constrained reaching movements. Advanced Robotics, 21(13): 1521-1544. https://doi.org/10.1163/156855307782148550

[20] Lowrey, K., Kolev, S., Dao, J., Rajeswaran, A., Todorov, E. (2018). Reinforcement learning for non-prehensile manipulation: Transfer from simulation to physical system. In 2018 IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR), pp. 35-42. https://doi.org/10.1109/SIMPAR.2018.8376268

[21] Kober, J., Oztop, E., Peters, J. (2011). Reinforcement learning to adjust robot movements to new situations. In Twenty-Second International Joint Conference on Artificial Intelligence, pp. 2650-2655. https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-441

[22] Silva, I.J., Perico, D.H., Costa, A.H.R., Bianchi, R.A. (2017). Using reinforcement learning to optimize gait generation parameters of a humanoid robot. XIII Simpósio Brasileiro de Automaçao Inteligente, pp. 288-294.

[23] Kim, S.K., Kirchner, E.A., Stefes, A., Kirchner, F. (2017). Intrinsic interactive reinforcement learning–Using error-related potentials for real world human-robot interaction. Scientific Reports, 7(1): 1-16. https://doi.org/10.1038/s41598-017-17682-7

[24] Sharma, R., Singh, I., Bharadwaj, D., Prateek, M. (2019). Incorporating forgetting mechanism in Q-learning algorithm for locomotion of bipedal walking robot. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 8(7): 1782-1787.

[25] Bharadwaj, D., Prateek, M. (2019). Kinematics and dynamics of lower body of autonomous humanoid biped robot. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 8(4): 141-146.

[26] Bharadwaj, D., Prateek, M., Sharma, R. (2019). cDevelopment of reinforcement control algorithm of lower body of autonomous humanoid robot. International Journal of Recent Technology and Engineering (IJRTE), 8(1): 915-919.